

Android Test Application

Requirements:

- This is an example application requirement. For this test only login view is required and optional one of the views in 'Screen 2' (Favourites, Messages or Contacts).
- The application should work in portrait mode only
- Supported android versions: 4 + (min API 15)
- Screen resolution: 1080x1920 (Equivalent to Nexus 5)

[Screen #1 - Welcome screen](#)

[Assets:](#)

[Screen consists of:](#)

[Login view:](#)

[Screen #2 - Main application screen](#)

[Assets:](#)

[Screen consists of:](#)

[Tab widget with three tabs.](#)

[Favourites](#)

[Messages](#)

[Contacts](#)

The application consist of 2 screens:

Screen #1 - Welcome screen

Assets:

- Logotype
- Font - Roboto (used in the logo title) - [here you can download the font from Android dev website](#)
 - The font is also used as main font for the application
- Background image - 1 image is enough to illustrate the idea

Screen consists of:

On this screen we have the logo of the application and the name listed under the logo, followed by the email and password fields, and the login button.

There is an image on the background with applied on top of it gradient across the whole screen from black fading to transparent.

The image and the gradient should be place under the login form and the logo.

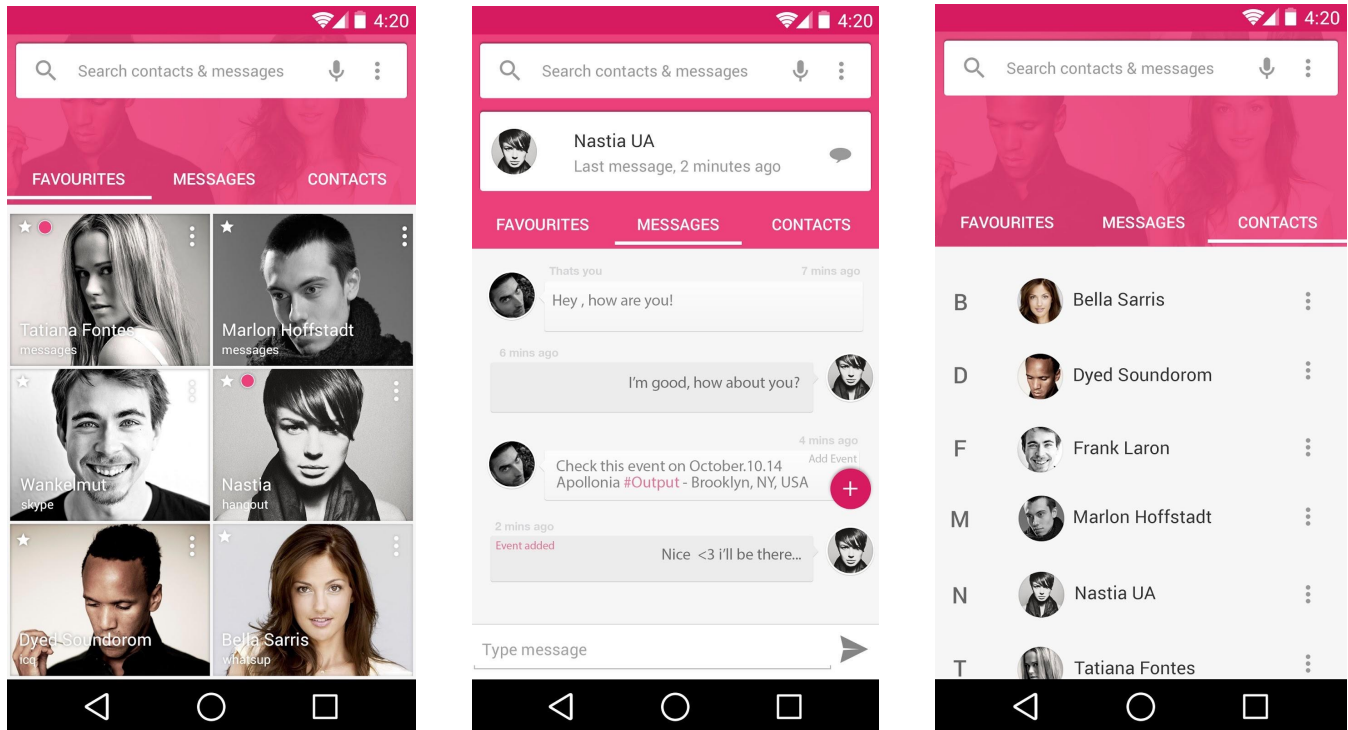
At the bottom of the screen we have a message asking the user to create and account if they don't have an existing one yet. This is not required for the test, meaning there is no need to open another view with registration form. It is for illustrative purposes only.



User flow:

When the user presses the “Log In” button the screen slides up and unveils the favourites tab on the main screen.

Screen #2 - Main application screen



Assets:

- For resources used in the application please visit [this address](#) containing graphic resources you might need for the visual interface. (Stencils, Icon Packs and Fonts)
- Custom resources are available in the **PSD file attached** to this task.

Screen consists of:

The screen starts with an search widget followed by a tab widget. Below is the grid view with all the friends tagged as favourite. Each entry consists of profile image, name, IM source, Star (indicating that the friend is being faved) and a context menu in the top right corner of each entry allowing the user to remove the friend from the fave list or start and chat with him.

The tab widget has three tabs.

- i. Tab one is favourites. **Grid view** with all faved friends.
- ii. Tab two is messages. **Listview** with custom ui
- iii. Tab three is contacts. **Table view** listed by default in alphabetical order. [First Name / Last Name]

Use the views specified above for each tab to achieve the user interface.

Favourites tab

On this screen we have a search widget. This widget should search within the contacts and messages databases and return results using a view similar to the one used in the third tab “Contacts” which is a list view. The context menu should be able to apply additional filter allowing the user to search only friends who are online at the moment.

Under this search widget (starting right after the status bar) we have a view showing a grid (**replica**) of the grid view below (the main grid with all your faved friends) without the extra metadata. (Just the images. No name, No favicon, No IM source and No context menu).

On top of this second view (the grid replica) we have a third view, which is set to be a solid color **#ec407a with opacity set to 90% to achieve the effect shown in the preview**

The search widget should always be on top of the other two views !!!

On swipe we want the (replica gridview to update in full sync with the gridview below, always showing the last 2 entries from the grid view. (they might not be the last entries in the the array, just the last two visible on the screen)

User flow:

Tab one - Favourites. When the user taps on a grid view entry we have to slide to the second tab “Messages” and start chat session.

When the user taps on the context menu of each grid view entry a context menu appears (use android default) with the following options:

- Remove from faves
- Add to group
- Send message

When the user taps on the image a profile view should load. **Do not** implement this view in the test application. For demonstration purposes please add more than 6 entries in the grid view (repeat the once you have with different names). 15 will be enough to illustrate the functionality of the view.

The chat tab consist of the following elements:

Each message entry has an avatar image, bubble (containing the message), metadata and event notifier. Above the bubble we list the timestamp for the message. (Format: 7 mins ago | 1 hour ago | 12 hours ago | Yesterday 9:53 am , etc)

Above the tab widget we have added an info card with this persons image and time stamp showing the last message posted in this conversation (Empty state is: user name and user status: offline | online).

On swipe up (more messages are needed on the screen) the tab widget moves up and sticks under the status bar. **The info card disappears from the view.**

Your messages are listed on the left hand side and your friends messages are listed on the right. Friends messages are with different background color.

If you decide to add an event posted by someone or the other user decides to do this the next bubble should represent this showing the "Event added" indication respectively in the top left or right corner depending on who added the event.

Adding event is an option icon in the keyboard view that opens another view allowing you to specify Date/ Time, location and #hashtag. **Do not** implement this view in the test application

Messages are all we need to illustrate the functionalities of the chat. On each message entered we need a dummy response message to illustrate a chat environment.

The contacts tab consist of the following:

A list of all the contacts you have listed alphabetically [First / Last name] and context menu on the right hand side allowing you to add this contact to your faves or delete it from the contacts.

Note if you have more than one person with first name starting with the letter B you have to list both of them under the letter B group, and while swiping you need to keep the letter visible until all contacts with names starting with this letter are swiped out of the visible canvas of the tab view. Similar to the contacts in your phonebook.

This concludes the test task requirements.

If you need more information feel free to contact us.

Email Address:

petar.gezenchov@1stonlinesolutions.com

The logo for First Online Solutions, featuring the text "First Online Solutions" in white, stacked vertically, on a blue square background with a slight shadow effect.

First
Online
Solutions